BIOS

From Wikipedia, the free encyclopedia.

In computing, the **Basic Input-Output System** or **BIOS** is computer interface code that locates and loads the operating system into RAM. It provides low-level communication, operation and configuration to the hardware of a system, which at a minimum drives the keyboard and provides primitive output to a display. The BIOS is usually written in Assembly language native to the processor.

The term first appeared in the CP/M operating system, describing the part of CP/M loaded during boot time that interfaced directly with the hardware. (CP/M machines usually had a simple boot loader in ROM, and nothing else.) Most versions of DOS have a file called "IBMBIO.COM" or "IO.SYS" that is analogous to the CP/M disk BIOS.

While the name BIOS is an acronym, it is also a play on the Greek word βιος (bios) *life*.

Contents
1 Boot BIOS
2 BIOS as firmware
3 Firmware on adapter cards
4 The BIOS boot specification
5 External links

Boot BIOS

Since the introduction of the IBM PC compatible in August 1981, BIOS code issues the first commands to the system during power-up or the boot process, such as where to find and how to load the operating system. PC BIOS code also contains diagnostics to assure critical hardware components, such as memory, keyboard, disk drive, I/O ports etc., are operational, and properly initialized. Nearly all BIOS implementations can optionally execute a setup program interfacing the CMOS Memory; this memory holds user-customizable configuration data (time, date, hard drive details, etc.) accessed by BIOS code. The 80x86 source code for early PC and AT BIOS was included with the IBM Technical Reference Manual.

In most modern BIOS implementations, one can select what boots first: CD, Hard disk, floppy disk, flash keydrive and so on. This is particularly useful for booting operating systems (e.g. Linux, especially Knoppix or Gnoppix) from a CD, without installing it in the hard disk, and for selecting the order of testing for the presence of bootable media.

Some BIOS systems allow the user to select the operating system to load (e.g. load Linux from the second hard disk), though this is more often handled by a second-stage boot loader.

BIOS as firmware

BIOS is sometimes called firmware because it is an integral part of the system hardware. Before 1990 or so BIOSs were held on ROM chips that could not be altered. As their complexity and the need for updates grew, BIOS firmware is stored on EEPROM or flash memory devices that can be upgraded by the user. However, an improperly executed or aborted BIOS update can render the computer or device unusable. To avoid BIOS corruption, some new motherboards have a backup BIOS. Also, most BIOSes have a "boot block" which is a portion of the ROM which runs first and is not updateable. This code will verify that the rest of the BIOS is intact

(via checksum, hash, etc.) before jumping to it. If the boot block detects that the main BIOS is corrupt, then it will typically boot to a floppy so that the user can try flashing again, hopefully with a better image. Hardware manufacturers frequently issue BIOS updates to upgrade their products and remove bugs.

Firmware on adapter cards

A computer system may contain several BIOS firmware chips. In addition to the boot BIOS, which contains code to access fundamental hardware components such as the keyboard or the floppy drive, plug-in adapter cards such as SCSI or USB hard disk adapters or video boards may include their own BIOS, complementing or replacing the system BIOS code for the given component.

To find these memory mapped expansion ROMs during boot, PC BIOS implementations scan real memory from 0xC8000 to 0xF0000 on 2 kilobyte boundaries looking for a 0x55 0xaa signature, which is immediately followed by a byte indicating the number of 512 byte blocks the expansion ROM occupies in real memory. The BIOS then jumps to the offset immediately after the size byte, at which point the expansion ROM code takes over and uses the BIOS services to provide a user configuration interface, register interrupt vectors for use by post-boot applications, or display diagnostic information.

The BIOS boot specification

If the expansion ROM wishes to change the way the system boots (such as from a network or a SCSI adapter the BIOS has no driver code for), it can use the BIOS Boot Specification (BBS) API to register its ability to do so. Once the expansion ROMs have registered using the BBS APIs, the user can select among the available boot options from within the BIOS's user interface. This is why most BBS compliant PC BIOS implementations will not allow the user to enter the BIOS's user interface until the expansion ROMs have finished executing and registering themselves with the BBS API.

See also : machine code, microchip, BIOS boot devices, BIOS Interrupt Calls.

External links

- Phoenix Technologies (*http://www.phoenix.com/*)
- American Megatrends Inc. (*http://www.ami.com/*)
- http://www.simtel.net/pub/pd/50185.html
- LinuxBios Standard (*http://www.linuxbios.org*)
- Linux Openbios Wiki (*http://www.openbios.org/openbioswiki*)
- BIOS Agent (*http://www.esupport.com/biosagent/*)
- The BIOS Boot Specification (*http://www.phoenix.com/NR/rdonlyres/56E38DE2-3E6F-4743-835F-B4A53726ABED/0/specsbbs101.pdf*)
- BNOBTC - BIOS Tool collection - (All?) BIOS Tools in one collection (*http://www.winamp5.de.tt/borg_bios/borgs_biostool-collection*)
- How to flash a BIOS (*http://www.pcnineoneone.com/howto/biosflash1.html*)

- This page was last modified 08:37, 21 Jan 2005.
- All text is available under the terms of the GNU Free Documentation License (see **Copyrights** for details).